IN THE CLAIMS

1. (Currently Amended) A method for processing digital data for use with a storage medium, comprising the steps of:

arranging ~~sequential~~ input digital data so as to form a plurality of data blocks of a predetermined matrix form;

appending outer parity of a predetermined size and inner parity of a predetermined size to each column and row of each of said plurality of data blocks, respectively~~, thereby forming one ECC (Error Correction Code) block to perform an error correction on the ECC block~~ , and combining the plurality of data blocks as appended into one ECC (Error Correction Code) block to perform an error correction on the basis of the ECC block;

reordering rows including the outer parity so as to insert said rows including the outer parity separately into other rows including no outer parity, for each of said plurality of data blocks in said ~~appending~~ combining step; and

writing sequentially rows [[in]] having the same ~~order~~ row number in said plurality of data blocks re-arranged in said reordering step, to the storage medium ~~sequentially on row-by-row basis~~.

2. (Currently Amended) The method according to claim 1, wherein in said arranging step, each of said plurality of data blocks is formed by using the following equations:

i=b/X and

j=b-(X×i),

where i and j (0≤i≤ (Y-1) and 0≤j≤ (X -1)) represent row and column position in each data block of (X×Y)-byte size, respectively,

where b (0≤b≤ (X×Y) -1) represents an order in which the bytes in said ~~sequential~~ input data of (X×Y)-byte size are inputted, and

where X and Y are integers greater than 1.

3. (Previously Presented) The method according to claim 2, wherein X is 172 and Y is 192.

4. (Previously Presented) The method according to claim 3, wherein said outer parity is 16-byte long in each column and said inner parity is 10-byte long in each row.

5. (Previously Presented) The method according to claim 1, wherein said appending step comprises of the sub-steps of:

appending said outer parity of a predetermined size to each column of each of said plurality of data blocks in the column direction; and

appending said inner parity of a predetermined size to each row of each of said plurality of data block outer-parity-encoded in said outer parity appending step in the row direction.

6. (Previously Presented) The method according to claim 1, wherein in said writing step, said plurality of data blocks include N ($N \geq 2$) data blocks, each being ($J \times K$)- bytes in size, and a byte at ($R(i)$, $C(i)$) of the i-th data block is selected and written to said storage medium based on the following equations:

$S=R(i)\times(J\times N)+C(i)+J\times(i-1)$, where ($R(i)$, $C(i)$) ($0 \leq R(i) \leq (K-1)$ and $0 \leq C(i) \leq (J-1)$) represents row and column position in the i-th data block, and s ($0 \leq S \leq (J\times K\times N)-1$) represents an order in which bytes in all data blocks sequentially are written to said storage medium.

7. (Previously Presented) The method according to claim 6, wherein J is 182 and K is 208.

8. (Currently Amended) A method for processing digital data for use with a storage medium, comprising the steps of:

arranging ~~a sequential~~ input digital data so as to form a pair of data blocks of a predetermined matrix form, wherein each of said pair of data blocks is formed by using the following equations:

$i=b/X$ and

$j=b-(X\times i)$,

where i and j ($0 \leq i \leq (Y-1)$ and $0 \leq j \leq (X-1)$) represent row and column position in each data block of ($X\times Y$)-byte size, respectively,

where b ($0 \leq b \leq (X\times Y)-1$) represents an order in which the bytes in said sequential input data of ($X\times Y$)-byte size are inputted, and where X is 172 and Y is 192; ~~and~~

appending an outer parity of a predetermined size and an inner parity of a predetermined size to each column and row of each of said plurality of data blocks, respectively; and

combining said plurality of data blocks as appended into ~~, thereby forming~~ one ECC (Error Correction Code) block to perform an error correction ~~on the~~ using the ECC block.

9. (Previously Presented) The method of claim 8, further comprising:

interleaving rows including the outer parity separately at a position after a predetermined row for each of said pair of data blocks in said appending step.

10. (Currently Amended) The method of claim 9, further comprising:

writing sequentially rows [[in]] having the same ~~order~~ row number in said pair of data blocks re-arranged in said interleaving step, to the storage medium ~~sequentially on row-by-row basis after modulating the digital data by a predetermined modulation algorithm~~.

11. (Previously Presented) The method of claim 10, wherein the predetermined modulation algorithm converts 8 bits to a given bit, the given bit being larger than 8 bits.

12. (Previously Presented) The method of claim 11, wherein said outer parity is 16-byte long in each column and said inner parity is 10-byte long in each row.

13. (Previously Presented) The method of claim 8, wherein said appending step comprises of the sub-steps of:

appending said outer parity of a predetermined size to each column of each of said pair of data blocks in the column direction; and

appending said inner parity of a predetermined size to each row of each of said pair of data blocks outer-parity-encoded in said outer parity appending step in the row direction.

14. (Currently Amended) A method for processing digital data for use with a storage medium, comprising the steps of:

arranging a sequential input digital data so as to form a pair of data units of a predetermined matrix form, each data unit having size of 172×192 bytes;

appending an outer parity of a predetermined size and an inner parity of a predetermined size to each column and row of each data unit, respectively; and

combining ~~each~~ the pair of data units having the appended outer parity and inner parity~~, thereby forming~~ into one ECC (Error Correction Code) block ~~to perform an error correction on the ECC block~~.

15. (Previously Presented) The method of claim 14, further comprising:

interleaving rows including the outer parity separately at a position after a predetermined row for each data unit in said combining step.

16. (Currently Amended) The method of claim 15, further comprising:

modulating the digital data by a predetermined modulation algorithm; and

writing the modulated data of rows [[in]] having the same ~~order~~ row number in said pair of data units re-arranged in said interleaving step, to the storage medium sequentially ~~on row-by-row basis~~.

17. (Previously Presented) The method of claim 16, further comprising:

appending a predetermined sync code to the modulated data in the modulating step, and then performing-the writing step of writing the sync code and the modulated data.

18. (Previously Presented) The method of claim 17, wherein the predetermined sync code of 4 columns is appended within the ECC block before the writing step.

19. (Previously Presented) The method of claim 14, wherein 16 rows including said outer parity are appended to each data unit and 10 columns including said inner parity are appended to each data unit.

20. (Currently Amended) The method of claim 19, wherein the one ECC block ~~consists~~ has a size of 364×208 bytes.

21. (Currently Amended) A storage medium having a data structure of ECC (Error Correction Code) block for an error correction,

wherein the ECC block is constructed by arranging ~~a sequential~~ input digital data so as to form a pair of data units of a predetermined matrix form, appending an outer parity of a predetermined size and an inner parity of a predetermined size to each column and row of each data unit, respectively, and combining ~~each~~ the pair of data units having the appended outer parity and inner parity~~, thereby forming~~ into one ECC block to perform an error correction ~~on~~ using the ECC block,

wherein each row including the outer parity separately is interleaved at a position after a predetermined row for each data unit, the digital data of each data unit is modulated by a predetermined modulation algorithm, and the modulated data of rows ~~is recorded in~~ having the same ~~order~~ row number in said pair of data units are sequentially recorded ~~on row-by-row basis~~.

22. (Previously Presented) The storage medium of claim 21, wherein the arranged each data unit has size of 172×192 bytes respectively.

23. (Previously Presented) The storage medium of claim 21, wherein a predetermined sync code is appended to the modulated data, and then the sync code and the modulated data are recorded together.

24. (Previously Presented) The storage medium of claim 23, wherein the predetermined sync code of 4 columns is appended within the ECC block before the recording of the modulated data.

25. (Previously Presented) The storage medium of claim 21, wherein 16 rows including said outer parity are appended to each data unit and 10 columns including said inner parity are appended to each data unit.

26. (Currently Amended) The storage medium of claim 25, wherein the one ECC block ~~consists of~~ has a size of 364×208 bytes.

27. (New) A storage medium comprising:

an error correction code (ECC) block stored on the storage medium, the ECC block having a size greater than 172×192 bytes and having rows of outer parity interleaved with rows without outer parity.

28. (New) The storage medium of claim 27, wherein the ECC block has a size of 364×208 bytes.

29. (New) The storage medium of claim 27, wherein the ECC block is constructed by combining a pair of ECC sub-blocks, the combining involving reading out sequentially rows having the same row number in the ECC sub-blocks and recording the read-out rows having the same row number as one row of the ECC block.

30. (New) The method of claim 14, wherein the combining step includes:

reading out sequentially rows having the same row number in the pair of data units; and

recording the read-out rows having the same row number as one row of the ECC block.